Penetration Testing

Penetration Testing

> A penetration test, or "pen test" is an attempt to evaluate the security of IT infrastructures using a controlled environment to safely attack, identify, and exploit vulnerabilities. These vulnerabilities may exist in operating systems, services, networks, and application. They may also exist due to improper configurations or risky end-user behavior.
>
> Penetration testing assessments are also useful in validating the efficacy of defensive mechanisms and determining how well end-users adhere to security policies. [^1]

Penetration Test Phases

1. Pre-engagement Interactions
2. Intelligence Gathering
3. Threat Modeling
4. Vulnerability Analysis
5. Exploitation
6. Post Exploitation
7. Reporting

1. Pre-engagement Interactions

- Scoping Meeting
- Questionaires for different scopes
  - Network, Web Application, WiFi, Physical, Social Engineering
- Framing conditions
  - Start & end date, IP ranges & domains, dealing with 3rd parties etc.
- Emergency Contact Information
- 📜 **Rules of Engagement**

2. Intelligence Gathering

- Choose appropriate level of Intelligence Gathering
  - Level 1: Compliance Driven (base minimum, mostly automated through tools)
  - Level 2: Best Practice (L1 + manual analysis e.g. business understanding, org chart, etc.)
  - Level 3: State Sponsored (L1 + L2 + heavy manual analysis e.g. cultivating relationships on social networks, deep understanding of business relationships, etc.)

Intelligence Gathering Aspects

- OSINT
- Covert Gathering (Corporate / HUMINT)
- Footprinting
- Identify Protection Mechanisms

3. Threat Modeling

- Business Asset Analysis (e.g. policies, product or financial data, technical information, employee or customer data)
- Business Process Analysis (including IT, HR or 3rd party support processes)
- Threat Agents/Community Analysis (see Attackers Exercise)
- Threat Capability Analysis (e.g. used tools or availability of relevant exploits)
- Motivation Modeling (e.g. direct vs. indirect profit, grudge, fun/reputation, springboard)

4. Vulnerability Analysis

> Vulnerability testing is the process of discovering flaws in systems and applications which can be leveraged by an attacker. These flaws can range anywhere from host and service misconfiguration, or insecure application design. Although the process used to look for flaws varies and is highly dependent on the particular component being tested, some key principals apply to the process.
>
> When conducting vulnerability analysis of any type the tester should properly scope the testing for applicable depth and breadth to meet the goals and/or requirements of the desired outcome.

> [...] Once a vulnerability has been reported in a target system, it is necessary to determine the accuracy of the identification of the issue, and to research the potential exploitability of the vulnerability within the scope of the penetration test.

Types of vulnerability analysis

- Active, e.g.
  - automated application scans
  - banner grabbing
  - network scans
- Passive (metadata analysis and traffic monitoring)

5. Exploitation

> The exploitation phase of a penetration test focuses solely on establishing access to a system or resource by bypassing security restrictions. If the prior phase, vulnerability analysis was performed properly, this phase should be well planned and a precision strike.

- Avoid or bypass countermeasures (e.g. AV, ASLR or WAF)
- Escape detection (e.g. evade IDS/IPS or avoid cameras)
- Precision Strike
- Tailored Exploits
- Zero-Day Angle

6. Post Exploitation

> The purpose of the Post-Exploitation phase is to determine the value of the machine compromised and to maintain control of the machine for later use. The value of the machine is determined by the sensitivity of the data stored on it and the machines usefulness in further compromising the network.

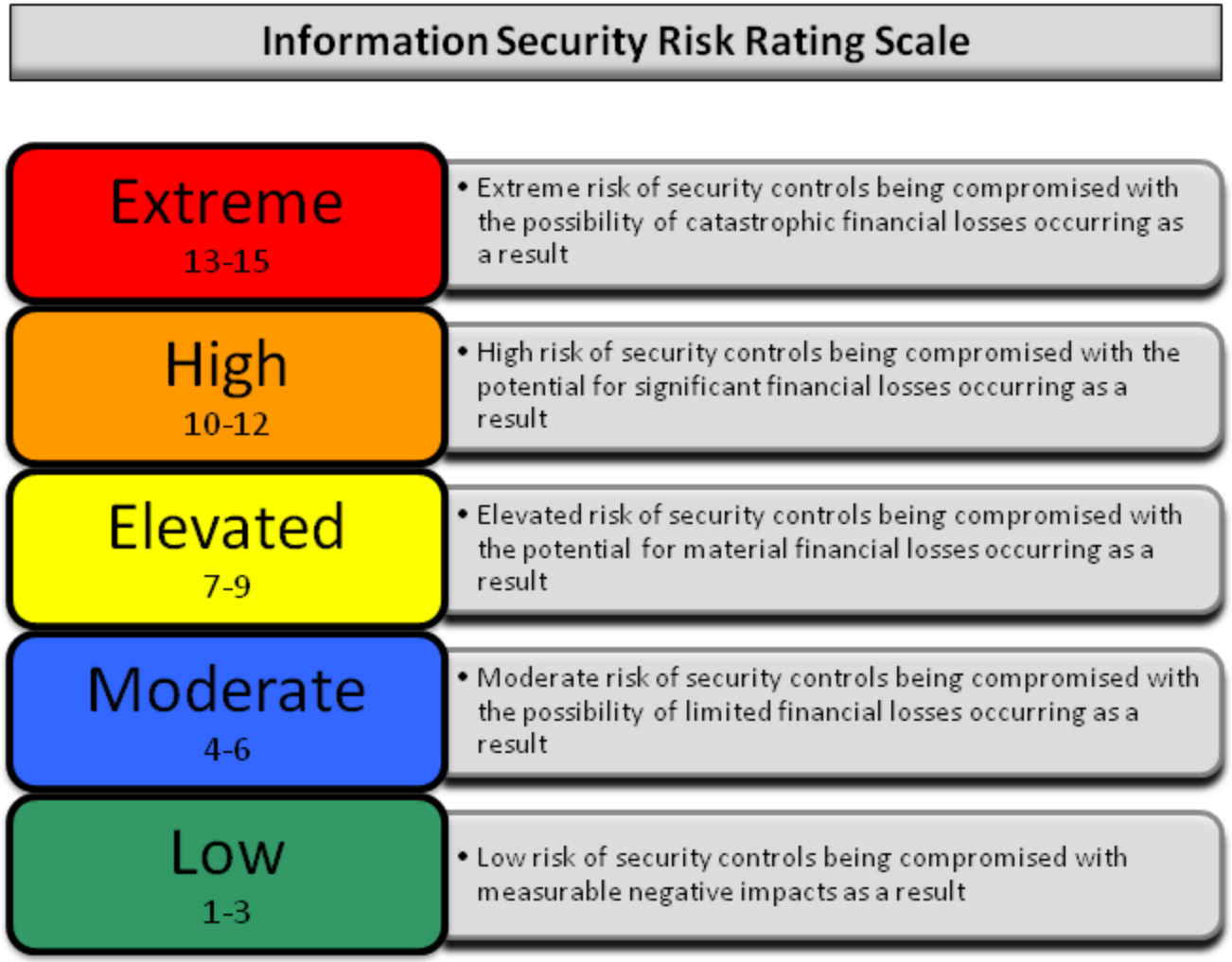
- Infrastructure Analysis (for potential targets deeper in the network)
- Pillaging (e.g. personal information, credit card information, passwords, etc.)
- Data Exfiltration
- Persistence (e.g. installation of backdoor or creating an alternate user account)
- Cleanup (after the penetration test has been completed)

7. Reporting

A typical pentest reports consists of two major sections in order to communicate the objectives, methods, and results of the testing conducted to various audiences.

1. **Executive Summary**: Specific goals of the Penetration Test and the high level findings of the testing exercise. The intended audience will be those who are in charge of the oversight and strategic vision of the security program as well as any members of the organization which may be impacted by the identified/confirmed threats.
2. **Technical Report**: Technical details of the test and all of the aspects/components agreed upon as key success indicators within the pre engagement exercise. The technical report section will describe in detail the scope, information, attack path, impact and remediation suggestions of the test.

Risk Rating Scale



Awesome Penetration Testing

> A collection of awesome penetration testing resources.

https://github.com/enaqx/awesome-pentest

Bug Bounty Programs

> A bug bounty program is a deal offered by many websites and software developers by which individuals can **receive recognition and compensation for reporting bugs, especially** those pertaining to **exploits and vulnerabilities**. These programs allow the developers to discover and resolve bugs before the general public is aware of them, preventing incidents of widespread abuse. [^2]

HackerOne's list of known Bug Bounty Programs

- https://hackerone.com/bug-bounty-programs

💰 *Bounties may range from recognition over swag up to >100.000$!*

Web Security Policies (`security.txt`)

> When security vulnerabilities are discovered by researchers, proper reporting channels are often lacking. As a result, vulnerabilities may be left unreported. This document defines a format ("security.txt") to help organizations describe their vulnerability disclosure practices to make it easier for researchers to report vulnerabilities.

Fields of `security.txt`

- **Contact**: An address that researchers should use for reporting security issues
- **Encryption**: Encryption key that security researchers should use for encrypted communication
- **Acknowledgements**: Link to a page where security researchers are recognized for their reports
- **Permission**: Value `none` indicates to security researchers that they must not perform any kind of testing. Must not be interpreted as having any legal value.

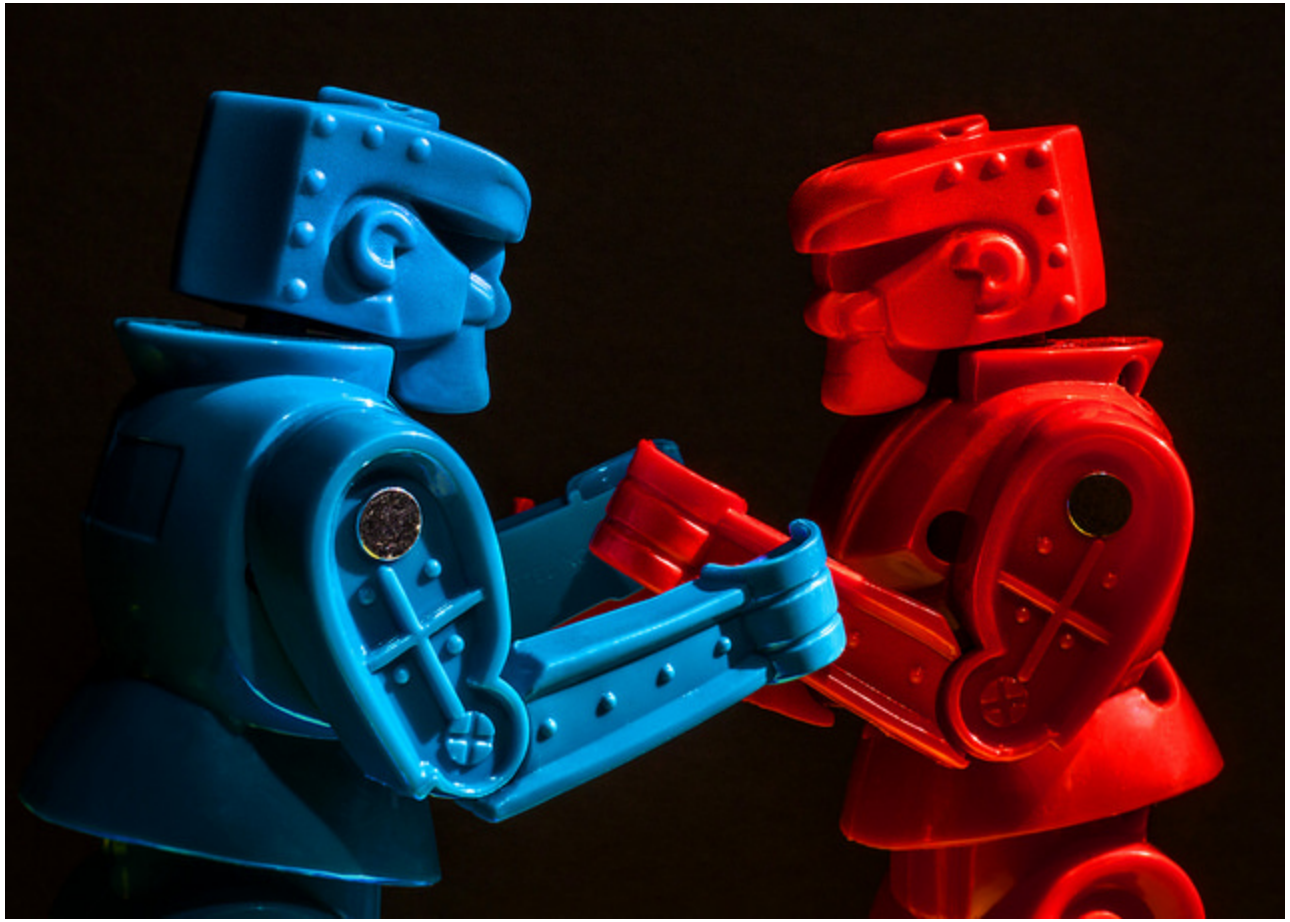
- **Policy**: Link to where your security policy and/or disclosure policy is located
- **Signature**: Full URI of an external signature file that can be used to check the authenticity of a `security.txt` file
- **Hiring**: Linking to the vendor's security-related job positions

---

Example: https://securitytxt.org/.well-known/security.txt

```
# If you would like to report a security issue
# you may report it to us on HackerOne.
Contact: https://hackerone.com/ed
Encryption: https://keybase.pub/edoverflow/pgp_key.asc
Acknowledgements: https://hackerone.com/ed/thanks
```

Red vs. Blue



🚩 Red Team

> **Red Teams** are external entities brought in to test the effectiveness of a security program. This is accomplished by emulating the behaviors and techniques of likely attackers in the most realistic way possible. The practice is similar, but not identical to, penetration testing, and involves the pursuit of one or more objectives.

Exercise 8.1 (🏠)

1. What do you think: In a fight between pirates and ninjas, who would win?
2. Read Penetration Test vs. Red Team Assessment

👮 Blue Team

> **Blue Teams** refer to the internal security team that defends against both real attackers and Red Teams. Blue Teams should be distinguished from standard security teams in most organizations, as most security operations teams do not have a mentality of constant vigilance against attack, which is the mission and perspective of a true Blue Team.
>
> A team of highly skilled individuals who conduct systematic examinations of IS or products to determine adequacy of security measures, to identify security deficiencies, to predict effectiveness of proposed security measures, and to confirm adequacy of such measures after implementation. [^3]

Project Zero (Google)

> Project Zero is the name of a team of security analysts employed by Google tasked with finding zero-day vulnerabilities.
>
> After finding a number of flaws in software used by many end-users [...] Google decided to form a full-time team dedicated to finding such vulnerabilities, not only in Google software but any software used by its users.
>
> Bugs found by the Project Zero team are reported to the manufacturer and only made publicly visible once a patch has been released or if 90 days have passed without a patch being released. [^4]

https://googleprojectzero.blogspot.com/